BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Pengertian sistem terbagi dua yaitu dilihat dari pendekatan yang menekankan pada prosedur dan pendekatan yang menekankan pada elemen atau komponennya.

Menurut [Jog05] pendekatan sistem yang menekankan pada prosedur mendefinisikan sistem sebagai : ”*jaringan kerja dan prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan sasaran tertentu*”.

Adapun pendekatan sistem yang menekankan pada elemen atau komponennya mendefinisikan sistem sebagai : ”*kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu*”.

Dari kedua pengertian di atas dapat disimpulkan bahwa pengertian sistem adalah kumpulan elemen-elemen atau jaringan kerja dan prosedur-prosedur yang saling berinteraksi satu sama lain untuk mencapai tujuan atau sasaran tertentu.

2.1.1 Karakteristik Sistem

Menurut [Jog05] suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu :

a. Komponen-komponen (*Components*)

   Suatu sistem terdiri dari kumpulan komponen atau subsistem yang saling berinteraksi atau bekerja sama satu sama lain untuk membentuk satu

kesatuan. Setiap komponen atau subsistem tersebut memiliki sifat-sifat dan fungsi tertentu yang mempengaruhi proses sistem secara keseluruhan.

b. Batas Sistem (*Boundary*)

Batas sistem yaitu daerah yang membatasi antara suatu sistem dengan lingkungan luarnya atau dengan sistem yang lainnya. Batas suatu sistem merupakan ruang lingkup (*scope*) dari sistem tersebut. Dan batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan.

c. Lingkungan Luar Sistem (*environments*)

Environments dari suatu sistem adalah apapun yang berada diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus dijaga dan dipelihara. Sedangkan lingkungan luar sistem yang bersifat merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

d. Penghubung Sistem (*Interface*)

Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran (*output*) dari satu subsistem akan menjadi masukan (*input*) untuk subsistem yang lain dengan melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

e. Masukan (*Input*)

Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh di dalam istem komputer, program adalah *maintaenance input* yang digunakan untuk mengoperasikan komputer dan data adalah *signal input* untuk diolah menjadi informasi.

f. Keluaran (*Output*)

Keluaran merupakan hasil dari energi yang diolah. Keluaran ini dapat berupa keluaran yang berguna atau keluaran yang tidak berguna. Sebagai contoh panas yang dikeluarkan dari sistem komputer merupakan keluaran yang tidak berguna dan merupakan sisa pembuangan. sedangkan informasi adalah keluaran yang berguna dan dibutuhkan.

g. Pengolah (*process*)

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan-laporan keuangan dan laporan-laporan yang dibutuhkan oleh manajemen.

h. Sasaran Sistem (*Objectives*)

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang

dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.1.2 Klasifikasi Sistem

Menurut [Jog05] sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya adalah sebagai berikut:

a. Sistem abstrak (*abstrak system*) dan sistem fisik (*phisical system*)

   Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologi. Adapun sistem fisik merupakan sistem yang ada secara fisik, misalnya sistem komputer.

b. Sistem alamiah (*natural* system) dan sistem buatan manusia (*Human Made System*).

   Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Misalnya sistem perputaran bumi. Adapun sistem buatan manusia adalah sistem yang dirancang oleh manusia. Misalnya, sistem buatan manusia yang melibatkan interaksi antara manusia dengan mesin yang disebut *human machine system.*

c. Sistem tertentu (*Deterministik System*) dan sistem tak tertentu (*Probabilistik System*).

   Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti, sehingga keluaran dari sistem dapat diramalkan. Misalnya sistem komputer. Adapun sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat di prediksi karena mengandung unsur probabilitas.

d. Sistem tertutup (*Closed System*) dan sistem terbuka (*Open System*).

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur dari pihak luarnya. Secara teoritis sistem tertutup ini ada, tetapi kenyataannnya tidak ada sistem yang sifatnya tertutup, yang ada hanyalah *relatively closed system* (secara relatif tertutup, tidak benar-benar tertutup). Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya. Karena sistemnya terbuka dan terpengaruh lingkungan luarnya, maka suatu sistem harus mempunyai suatu sistem pengendalian yang baik.

2.2 Konsep Dasar Informasi

Menurut [Jog05] informasi adalah **”***data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya***”.**

Menurut [Kad03] informasi adalah *”data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang”*.

Adapun informasi yang dihasilkan dari proses pengolahan data tersebut pada kenyataannya memiliki kualitas yang berbeda. Menurut [Jog05] kualitas informasi tergantung dari tiga hal yaitu informasi harus akurat (*accurate*), tepat waktu (*timeliness*) dan relevan (*relevance*).

a. Akurat

Berarti informasi harus bebas dari kekurangan atau kesalahan-kesalahan. Akurat juga berarti informasi harus jelas mencerminkan keadaan yang sebenarnya.

b. Tepat waktu

Berarti informasi harus selalu ada pada saat dibutuhkan pengguna dan tidak datang terlambat. Karena informasi merupakan landasan di dalam pengambilan keputusan. Jadi bila terlambat, informasi tersebut tidak mempunyai nilai lagi dan dapat berakibat fatal bagi organisasi.

c. Relevan

Informasi untuk setiap orang satu dengan yang lainnya berbeda. Maka informasi yang relevan adalah informasi yang berguna dan sesuai dengan kebutuhan setiap pemakainya.

2.3 Konsep Dasar Sistem Informasi

Menyangkut pemahaman tentang pengertian sistem informasi ini, dalam bukunya [Kad03] mengutip beberapa pendapat para ahli, diantaranya :

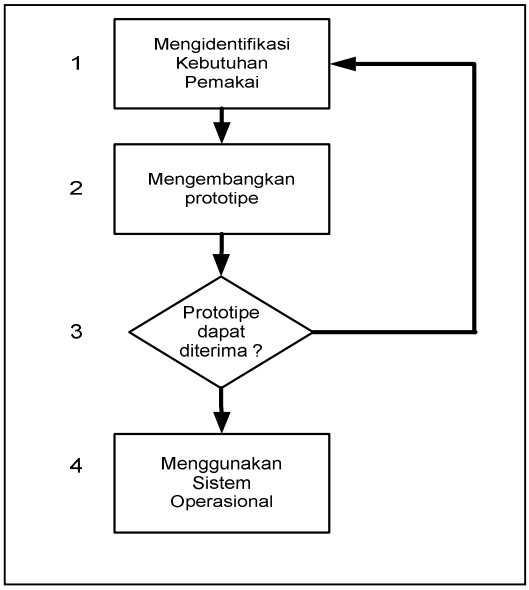
Menurut Hall sistem Informasi adalah ”*sebuah rangkaian prosedur formal di mana data dikelompokkan, diproses menjadi informasi dan didistribusikan kepada pemakai*”.

Menurut Bodnar dan Hopwood sistem Informasi adalah **”***Kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data kedalam bentuk informasi yang berguna***”.**

Dari pengertian di atas dapat disimpulkan bahwa yang dimaksud sistem informasi sesungguhnya tidak harus melibatkan komputer ketika proses mengolah data menjadi sebuah informasi. Adapun Sistem informasi yang melibatkan komputer biasa disebut sistem informasi berbasis komputer (*Computer-Based Information Systems* atau CBIS). Akan tetapi, pada umumnya kata sistem informasi diartikan sebagai sistem informasi berbasis komputer, meskipun tidak ditambahkan kata berbasis komputer.

2.4 Metode Pengembangan Sistem

Dewasa ini, metode pengembangan sistem untuk menyelesaikan sebuah proyek sistem informasi banyak sekali ragamnya, dan salah satunya adalah model prototipe. Metode ini bertujuan untuk mendapatkan kebutuhan yang jelas dan disetujui calon pemakai. Adapun tahapan dari model prototipe dapat dilihat pada gambar di bawah ini :



Gambar 2.1 Prototype Model

(Sumber : Raymond McLeod, George Schell, “Sistem Informasi Manajemen”)

Menurut [Kad03], sasaran prototipe secara garis besar adalah sebagai berikut:

1. Menyediakan umpan balik yang cepat dari pemakai (*user*) kepada pengembang (*system developer*).
2. Membantu menggambarkan kebutuhan pemakai (*user*) dengan kesalahan yang lebih sedikit.
3. Meningkatkan pemahaman pengembang (*system developer*) dan pemakai (*user*) terhadap sasaran yang seharusnya dicapai oleh sistem.
4. Menjadikan keterlibatan pemakai (*user*) sangat berarti dalam analisis dan desain sistem.

Seluruh metode pengembangan memiliki kelebihan dan kekurangan, berikut adalah kelebihan-kelebihan dan kekurangan-kekurangan dari model prototipe.

Berikut ini kelebihan model prototipe yaitu :

1. Pendefinisian kebutuhan pemakai menjadi lebih baik karena keterlibatan pemakai yang lebih intensif.
2. Memperkecil kesalahan disebabkan pada setiap versi prototipe kesalahan segera terdeteksi oleh pemakai.
3. Pemakai mempunyai kesempatan dalam meminta perubahan-perubahan.
4. Mempersingkat waktu dalam mengembangkan sistem secara keseluruhan.
5. Menghemat biaya jika dibandingkan dengan menggunakan metode SDLC tradisional.

Adapun kekurangan metode prototipe yaitu :

1. Sistem akan baik jika pemakai sungguh-sungguh meluangkan waktunya untuk menggarap prototipe.
2. Dokumentasi sering terabaikan karena pengembang lebih berkonsentrasi pada tahap pengujian dan pembuatan prototipe.
3. Waktu yang singkat dapat menghasilkan sistem yang tidak lengkap dan kurang teruji.
4. Jika proses pengulangan terlalu sering dapat mengakibatkan pemakai jenuh dan memberikan respon yang negatif.
5. Apabila prototipe tak dikelola dengan baik dapat mengakibatkan prototipe tak pernah berakhir karena usulan perubahan terlalu sering dipenuhi.

2.5 Alat Pengembangan Sistem

Dalam mengerjakan tahapan-tahapan yang berada dalam metode pengembangan sistem, maka dibutuhkan alat agar lebih mudah dalam melaksanakannya. Adapun alat yang digunakan umumnya berupa suatu gambar diagram atau grafik, karena gambar dipandang akan lebih mudah untuk dipahami.

Adapun alat-alat yang digunakan dalam pengembangan sistem adalah sebagai berikut :

2.5.1 Diagram Konteks

Menurut [Lad05] diagram konteks adalah " *diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem*".

Suatu diagram konteks selalu mengandung satu dan hanya satu proses saja. Proses ini mewakili proses dari seluruh sistem dan tidak boleh ada *data store* dalam diagram konteks.

2.5.2 Data Flow Diagram

Menurut [Lad05] DFD adalah " *model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil*".

DFD ini sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik tempat data tersebut mengalir (misalnya lewat telepon, surat dan sebagainya), atau tempat data tersebut akan disimpan (misalnya hard disk, file kartu, diskette dan lain sebagainya).

Beberapa notasi atau simbol yang digunakan dalam DFD adalah sebagai berikut :

1. Kesatuan luar (*external entity*) disimbolkan dengan suatu notasi kotak Entitas ini merupakan kesatuan diluar sistem yang dapat berupa orang, organisasi, atau sistem lainnya yang berada di lingkungan luar lainnya yang akan memberikan input atau menerima output dari sistem.



Gambar 2.2 Notasi Entitas

(Sumber : Al-bahra Bin Ladjamudin "Analisis dan Desain Sistem Informasi")

2. Arus data (*data flow*) disimbolkan dengan suatu notasi panah. Notasi panah ini menggambarkan aliran data yang mengalir diantara proses, simpanan data,

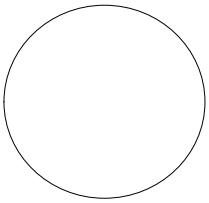
dan entitas baik berupa masukan untuk sistem ataupun hasil dari proses sistem.



Gambar 2.3 Notasi Arus Data

(Sumber : Al-bahra Bin Ladjamudin “Analisis dan Desain Sistem Informasi”)

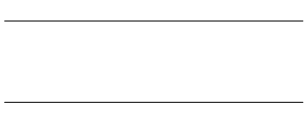
3. Proses (*process*) disimbolkan dengan suatu notasi lingkaran. Proses merupakan apa yang dikerjakan oleh sistem. Proses dapat mengolah data atau aliran data masuk menjadi aliran data ke luar. Proses berfungsi mentransformasikan masukan menjadi satu atau beberapa data keluaran sesuai dengan spesifikasi yang diinginkan



Gambar 2.4 Notasi Proses

(Sumber : Al-bahra Bin Ladjamudin “Analisis dan Desain Sistem Informasi”)

4. Simpanan Data disimbolkan dengan dua garis sejajar. Simpanan data merupakan tempat penyimpanan data pengikat data yang ada dalam sistem.



Gambar 2.5 Notasi Simpanan Data

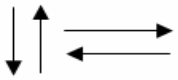
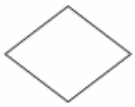
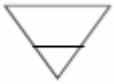
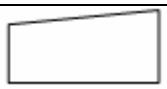
(Sumber : Al-bahra Bin Ladjamudin “Analisis dan Desain Sistem Informasi”)

2.5.3 Bagan Alir Dokumen (*Dokumen Flowchart*)

Bagan Alir Dokumen *(Dokumen Flowchart)* merupakan bagan alir yang menggambarkan arus dokumen-dokumen dan laporan-laporan termasuk tembusan-tembusannya pada sebuah sistem [Jog05].

Adapun simbol-simbol yang sering digunakan dalam membuat Bagan Alir Dokumen dapat dilihat pada tabel di bawah ini.

Tabel 2.1 Simbol-simbol Dokumen Flowchart

Simbol	Nama	Keterangan
	Dokumen	Menunjukan dokumen input dan output baik untuk proses manual atau komputer
	Proses manual	Menunjukan kegiatan atau pekerjaan manual
	Garis alir	Menunjukan arus atau arah liran dokumen atau data
	Proses	Menunjukan kegiatan proses dari operasi program komputer
	Hard disk	Menunjukan input atau output menggunakan hard disk
	Keputusan	Digunakan untuk sesuatu penyeleksian kondisi
	Simpanan offline	Digunakan untuk menyimpan data sebagai arsip secara manual
	Simbol Keyboard	Menunjukan Input menggunakan Keyboard
	Penghubung	Menunjukan penghubung ke halaman yang masih sama atau ke halaman lain
	Penjelasan	Digunakan untuk membuat keterangan mengenai suatu proses, supaya lebih jelas

2.5.4 Kamus Data

Menurut [Jog05] kamus data atau *data dictionary* adalah *" katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi "*.

Kamus data digunakan untuk mendefinisikan data yang mengalir dalam sebuah sistem secara lengkap berdasarkan arus data yang ada dalam DFD. Pada tahap analisis, kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem. Sedangkan pada perancangan sistem, kamus data dapat digunakan untuk merancang database , input dan laporan-laporan.

Adapun hal-hal yang perlu dicatat di kamus data adalah sebagai berikut :

1. Nama arus data

   Nama arus data perlu dicatat dalam kamus data yaitu untuk memudahkan mereka yang membaca DFD dan memerlukan penjelasan lebih lanjut tentang suatu arus data tertentu di DFD, sehingga dapat langsung mencarinya dengan mudah di kamus data.

2. Alias

   Alias atau nama lain dari data perlu dituliskan apabila nama lain ini ada. Hal ini karena untuk data yang sama kadang-kadang mempunyai nama yang berbeda untuk orang atau departemen lainnya. Sehingga manfaat penggunaan alias ini yaitu pembuatan dokumentasi menjadi lebih sederhana karena tidak adanya pengulangan dalam pencatatan kamus data untuk data yang sebenarnya sama.

3. Bentuk data

   Bentuk data ini perlu dicatat di kamus data karena dapat digunakan untuk mengelompokan kamus data kedalam kegunaannya sewaktu perancangan sistem. Adapun bentuk data yang mengalir dalam DFD dapat berupa dokumen dasar atau formulir, dokumen hasil cetakan komputer, laporan tercetak, tampilan layar monitor, field.

   Kamus data yang mencatat data yang mengalir dalam bentuk dokumen dasar atau formulir dapat digunakan untuk merancang bentuk input sistem. Kamus data yang mencatat data yang mengalir dalam bentuk laporan tercetak dan dokumen hasil cetakan komputer dapat digunakan untuk merancang output yang akan dihasilkan oleh sistem. Kamus data yang mencatat data yang mengalir dalam bentuk tampilan monitor dapat digunakan untuk merancang tampilan layar yang akan dihasilkan oleh sistem. Dan Kamus data yang mencatat data yang mengalir dalam bentuk-bentuk tersebut akan digunakan untuk merancang database.

4. Arus data

   Arus data ini menunjukan dari mana data mengalir dan mau kemana data akan menuju. Arus data ini perlu dicatat dalam kamus data untuk memudahkan pencarian arus data di DFD.

5. Penjelasan

   Penjelasan dapat diisi dengan keterangan-keterangan untuk lebih memperjelas makna dari arus data yang dicatat di kamus data.

6. Periode

   Periode ini menunjukan kapan terjadinya arus data ini. Periode ini perlu dicatat karena dapat digunakan untuk mengidentifikasi kapan input data akan dimasukan ke sistem, kapan proses proses dari program harus dilakukan dan kapan laporan-laporan harus diselesaikan.

7. Volume

   Volume yang perlu dicatat adalah volume rata-rata dan volume puncak dari arus data pada periode tertentu. Volume ini digunakan untuk mengidentifikasi besarnya simpanan luar yang akan digunakan, kapasitas dan jumlah alat input, alat pemroses dan alat output.

8. Struktur data

   Struktur data perlu dicatat dalam kamus data untuk menunjukan item-item data apa saja yang terdapat dalam suatu arus data.

Nama Arus Data : Nota Penjualan
Alias : Data penjualan
Bentuk Data : Dokumen dasar atau formulir
Arus Data : Proses 1.3 → Pembeli
Penjelasan : Sebagai bukti pembelian obat oleh konsumen
Periode : Harian
Volume : Volume rata-rata antara 40 – 150 Pembeli
Struktur Data :

No	**Nama Atribut**	**Keterangan**
1.	Tgl_Transaksi	Tanggal Transaksi
2.	No_Nota	No Nota
3.	Nama_Pembeli	Nama Pembeli
4.	Nama_Obat	Nama Barang
5.	Harga	Harga
6.	Jml_Obat	Jumlah Obat

Gambar 2.6 Contoh Kamus Data

2.5.5 Normalisasi

Menurut [Jog05] normalisasi adalah "*proses untuk mengorganisasikan file untuk menghilangkan grup elemen yang berulang-ulang*".

Menurut [Lad05] normalisasi adalah "*proses pengelompokan data kedalam bentuk tabel, relasi atau file untuk menyatakan entitas dan hubungan mereka sehingga terwujud satu bentuk database yang mudah untuk dimodifikasi***".**

Adapun langkah-langkah pembentukan normalisasi adalah sebagai berikut :

1. Bentuk Tidak Normal (Unnormalized Form)

   Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti format tertentu. Pada bentuk ini masih masih terdapat kerangkapan data atau redudansi.

2. Bentuk normal satu (*First normal form*) / 1 NF

   Bentuk normal satu yaitu bila relasi tersebut tidak memiliki atribut yang berulang atau bernilai ganda. Hal tersebut dapat dilakukan dengan menghilangkan semua atribut yang redudansi dari tabel yang tidak normal atau merubah seluruh atribut yang *multivalue* menjadi atribut yang *singlevalue*.

3. Bentuk normal dua (*Second normal form*) / 2NF

   Bentuk normal dua yaitu bila relasi tersebut memenuhi kriteria bentuk normal kesatu dan semua atribut bukan kunci bergantung sepenuhnya ke atribut kunci.

4. Bentuk normal tiga (*Third normal form*) / 3NF

   Bentuk normal tiga yaitu bila relasi tersebut memenuhi kriteria bentuk normal kedua dan atribut bukan kunci tidak boleh memiliki ketergantungan fungsional terhadap atribut bukan kunci lainnya. Seluruh atribut bukan kunci pada suatu

relasi hanya memiliki ketergantungan fungsional terhadap primary key pada relasi itu saja.

2.5.6 ERD (*Entity Relationship Diagram*)

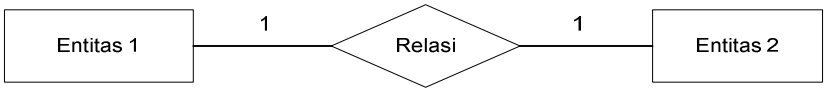
Menurut [Lad05] *Entity Relationship Diagram* adalah "*notasi grafik dari sebuah model data atau sebuah model jaringan yang menjelaskan tentang data yang tersimpan (Storgae data) dalam sistem secara abstrak*".

Entity Relationship Diagram tidak menyatakan bagaimana memanfaatkan data, membuat data, mengubah data dan menghapus data. Diagram ini menekankan pada struktur-struktur dan relationship data.

Dalam ERD terdapat elemen yang disebut dengan Kardinalitas Relasi. Menurut [Lad05] kardinalitas relasi menunjukan jumlah maksimum tupel yang dapat berelasi dengan entitas pada entitas lain. Terdapat tiga macam kardinalitas relasi, yaitu :

1. *One to One* (Satu ke Satu)

Tingkat hubungan satu ke satu dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya. Berdasarkan penjelasan di atas, ERD dengan kardinalitas relasi satu ke satu dapat digambarkan sebagai berikut :

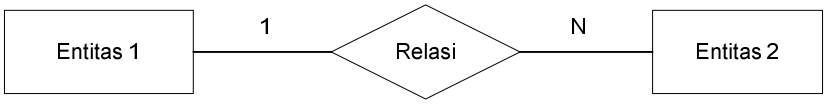


Gambar 2.7 Diagram Kardinalitas Satu ke Satu

2. *One to Many atau Many to One* (Satu ke Banyak atau Banyak ke Satu)

Tingkat hubungan satu ke banyak adalah sama dengan banyak ke satu. Tergantung dari arah mana hubungan tersebut dilihat. Untuk satu kejadian pada

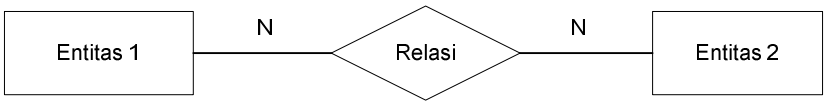
entitas pertama, dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua. Sebaliknya satu kejadian pada entitas kedua hanya dapat mempunyai satu hubungan dengan satu kejadian pada entitas yang pertama. Berdasarkan penjelasan di atas, ERD dengan kardinalitas relasi satu ke banyak atau banyak ke satu dapat digambarkan sebagai berikut :



Gambar 2.8 Diagram Kardinalitas Satu ke Banyak

3. Many to Many (Banyak ke Banyak)

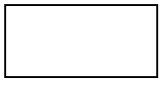
Tingkat hubungan banyak ke banyak terjadi jika setiap kejadian pada entitas pertama mempunyai banyak hubungan dengan kejadian pada entitas yang kedua dan sebaliknya. Berdasarkan penjelasan di atas, ERD dengan kardinalitas relasi banyak ke banyak dapat digambarkan sebagai berikut :

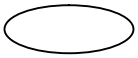
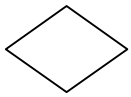


Gambar 2.9 Diagram Kardinalitas Banyak ke Banyak

Adapun simbol ERD dapat dilihat pada berikut ini :

Tabel 2.2 Simbol-simbol ERD (*Entity Relationship Diagram*)
(Sumber : Al-bahra Bin Ladjamudin "Analisis dan Desain Sistem Informasi")

Gambar/Simbol	Nama	Keterangan
	Entitas	Sesuatu apa saja yang ada di dalam sistem, nyata maupun abstrak. Entitas dapat berupa orang, benda, lokasi atau kejadian.

	Atribut	Sifat atau karakteristik dari setiap entitas maupun relationship.
	Belah Ketupat	Hubungan alamiah yang terjadi antar entitas.
	Link/Garis	Penghubung antara relasi dengan entitas dan entitas dengan atributnya.

2.5.7 Relasi Tabel

Relasi tabel adalah gambaran tentang hubungan yang terjadi antar tabel – tabel yang akan digunakan dalam program aplikasi melalui teknik normalisasi sehingga memiliki sebuah kunci yang menghubungkan relasi datanya.

2.6 Arsitektur Jaringan

Pada bagian ini akan dijelaskan mengenai pengertian jaringan komputer, jenis-jeni jaringan komputer dan topologi jaringan komputer.

2.6.1 Pengertian Jaringan Komputer

Menurut [Kad03] jaringan komputer adalah "*hubungan dua buah simpul (umumnya berupa komputer) atau lebih yang tujuan utamanya adalah untuk melakukan pertukaran data*".

Dalam prakteknya, jaringan komputer memungkinkan untuk melakukan berbagi (*sharing*) perangkat lunak, perangkat keras, dan bahkan berbagi (*sharing*) kekuatan pemrosesan.

2.6.2 Jenis-jenis Jaringan Komputer

Menurut [Bud05] ada empat kategori utama jaringan komputer, yaitu :

1. LAN (*Local Area Network*)

LAN digunakan untuk menghubungkan komputer yang berada dalam suatu area yang kecil, jarak antara komputer yang dihubungkannya hanya bisa mencapai 5 sampai 10 km. Suatu LAN biasanya bekerja pada kecepatan 10 sampai 100 Mbps. LAN menjadi populer karena memungkinkan banyak pengguna untuk memakai sumber daya secara bersama-sama.

2. MAN (*Metropolitan Area Network*)

MAN merupakan suatu jaringan yang cakupannya meliputi suatu kota. MAN menghubungkan LAN-LAN yang lokasinya berjauhan. Jangkauan MAN bisa mencapai 10 sampai beberapa ratus km. Suatu MAN biasanya bekerja pada kecepatan 1,5 sampai 150 Mbps.

3. WAN (*Wide Area Network*)

WAN dirancang untuk menghubungkan komputer-komputer yang terletak pada suatu cakupan geografis yang luas, seperti hubungan dari satu kota ke kota lain dalam suatu negara. Cakupan WAN dapat meliputi 100 sampai 1.000 km, dan kecepatan antar kota bisa bervariasi antara 1,5 Mbps sampai 2,4 Gbps.

4. GAN (*Global Area Network*)

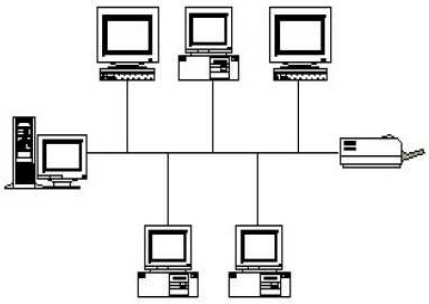
GAN merupakan jaringan yang menghubungkan negara-negara di seluruh dunia. Kecepatan GAN bervariasi mulai dari 1,5 Mbps sampai 100 Gbps dan cakupannya mencapai ribuan kilometer.

2.6.3 Topologi Jaringan Komputer

Menurut [Bud05] topologi fisik jaringan yang digunakan dalam jaringan diantaranya:

1. *Linear Bus* (Garis Lurus)

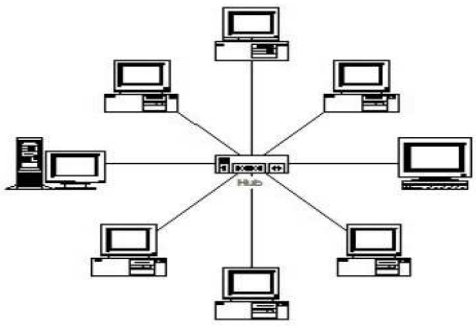
Topologi *linear bus* terdiri dari satu jalur kabel utama dimana masing-masing ujungnya diberikan sebuah *terminator*. Semua *nodes* pada jaringan (*file server*, *workstation*, dan perangkat lainnya) terkoneksi sebuah kabel utama (*backbone*).



Gambar 2.10 Topologi *Linear Bus*

2. *Star* (Bintang)

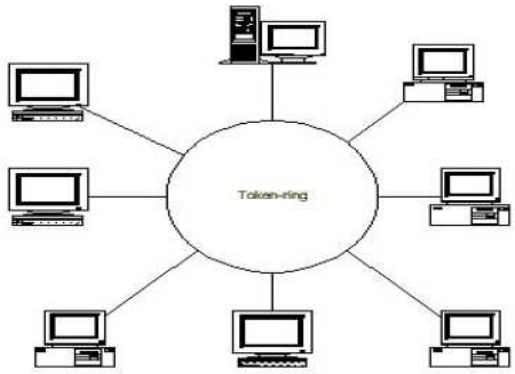
Pada topologi *star*, setiap *nodes* (*file server*, *workstation*, dan perangkat lainnya) terkoneksi ke jaringan melalui sebuah *concentrator*.



Gambar 2.11 Topologi *Star*

3. *Ring* (Cincin)

Topologi *ring* menggunakan teknik konfigurasi yang sama dengan topologi star tetapi pada topologi ini terlihat bahwa jalur media transmisi menyerupai suatu lingkaran tertutup.



Gambar 2.12 Topologi *Ring*

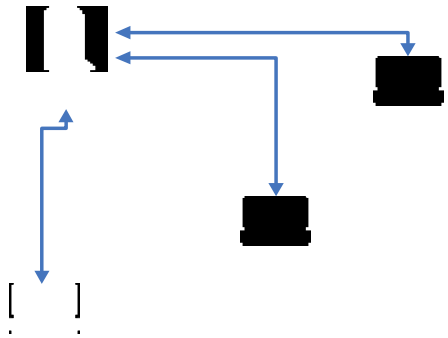
***4. Tree* (Pohon)**

Topologi model ini merupakan perpaduan antara topologi *linear bus* dan *star*, yang terdiri dari kelompok-kelompok dari *workstation* dengan konfigurasi *star* yang terkoneksi ke kabel utama yang menggunakan topologi *linear bus*. Topologi ini memungkinkan untuk perkembangan jaringan yang telah ada dan memungkinkan untuk mengkonfigurasi jaringan sesuai dengan kebutuhan.

2.7 Pengertian Client Server

Client-server merupakan model hubungan sebuah jaringan yang memungkinkan untuk mensentralisasi fungsi dari aplikasi kepada satu atau dua *dedicated file server*. Model hubungan ini menyediakan mekanisme untuk mengintegrasikan seluruh komponen yang ada di jaringan dan memungkinkan banyak pengguna secara bersamaan memakai sumber daya pada *file server.*

Sebuah *file server* menjadi jantung dari keseluruhan sistem, dan *Workstation* yang berdiri sendiri dapat mengambil sumber daya yang ada pada *file server*.



Gambar 2.13 Model hubungan *Client-Server*

2.8 Perangkat Lunak Pendukung

Adapun perangkat lunak yang digunakan untuk membangun sistem informasi ini yaitu Borland Delphi 7.0 dan Sql Server 2000. Berikut ini penjelasan singkat dari kedua perangkat pendukung tersebut :

2.8.1 Sekilas Tentang Borland Delphi 7.0

Borland delphi merupakan sarana pemrograman visual (*visual programming*). Bahasa pemrograman yang digunakan adalah bahasa pemrograman pascal dan kemudian disebut bahasa pemrograman delphi. Delphi merupakan penerus dari Turbo Pascal. Turbo Pascal dirancang untuk dijalankan pada sistem operasi DOS. Sedangkan delphi yang diluncurkan pertama kali tahun 1995 dirancang untuk beroperasi di bawah sistem Windows.

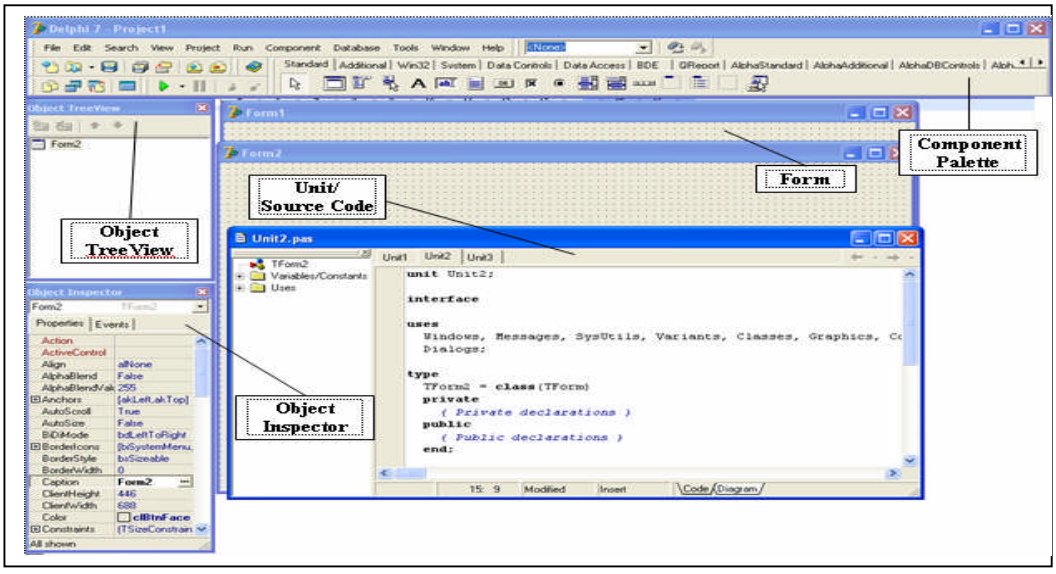
Kebutuhan akan adanya program aplikasi yang bekerja di bawah sistem operasi windows serta memiliki antarmuka visual yang *user friendly* telah

memancing minat banyak orang untuk menggunakan bahasa yang mampu menyediakan antarmuka grafis (*Graphical Use Interface*, GUI).

Dalam membuat aplikasi menggunakan Borland Delphi, akan sering ditemukan istilah-istilah yang biasa dipakai dalam lingkungan pemrograman visual, yaitu objek, properti, dan *event*. Delphi menerapkan konsep aplikasi yang digerakan oleh *event* (*event driven*). Pemrograman dengan model ini memang telah ada sebelum GUI dibuat dan diimplementasikan dalam beberapa cara. Tetapi teknik ini menjadi lebih berkembang dan banyak disukai oleh para programmer dan pemakai aplikasi semenjak diperkenalkannya mouse.

Pemrograman *event-driven* mencoba melengkapi kekurangan pemrograman prosedural dengan kerangka yang membedakan antara antarmuka pemakai dengan proses tertentu dalam aplikasi. Dengan adanya sarana pemrograman visual yang *event-driven* para pembuat aplikasi sangat terbantu ketika menyediakan sarana antarmuka bagi pemakai. Sehingga diharapkan akan lebih berkonsentrasi pada penanganan masalah aplikasinya bukan antarmukanya.

Lingkungan kerja Borland Delphi dapat dilihat pada gambar dibawah ini :



Gambar 2.14 Interface Borland Delphi 7.0

Fungsi dari elemen-elemen di atas adalah :

1. Object inspector : Suatu window yang berguna untuk mengatur suatu object baik properties, events dan method.
2. Form : Suatu window yang digunakan sebagai lembar kerja. di form ini semua komponen seperti tombol dan komponen lainnya disimpan.
3. Unit / Source code : Suatu window yang berisi perintah-perintah yang akan dieksekusi oleh computer. Pada layer inilah kita mengisikan program-program.
4. Component Palette : Suatu window yang berisikan komponen-komponen yang dipakai dalam program.
5. Object TreeView : Suatu windows berisi diagram pohon yang memperlihatkan komponen-komponen yang disimpan form data modul dan sebagainya.

2.8.2 Sekilas Tentang SQL Server 2000

SQL Server 2000 diluncurkan pada tahun 2000. SQL Server ini merupakan pengembangan dari versi sebelumnya yaitu SQL Server versi 7.0 yang diluncurkan tahun 1999. Microsoft SQL Server 2000 merupakan salah satu *database relational* yang banyak digunakan oleh dunia usaha baik perusahaan-perusahaan besar maupun menengah. SQL Server 2000 ini dirancang untuk penggunaan aplikasi dengan arsitektur *client-server*. Sehingga dengan menggunakan SQL server memungkinkan pengguna dapat mengolah data dalam database terpusat atau yang sering disebut sebagai server.

2.9 Pengertian Apotek

Dalam peraturan pemerintah nomor 25 tahun 1980 yang dimaksud apotek adalah suatu tempat tertentu, tempat dilakukan pekerjaan kefarmasian dan penyaluran obat kepada masyarakat [HKS05].

Adapun tugas dan fungsi apotek adalah sebagai tempat pengabdian profesi seorang apoteker yang telah mengucapkan sumpah jabatan, sarana farmasi yang melaksanakan peracikan, pengubahan bentuk, pencampuran dan penyerahan obat atau bahan obat, dan sarana penyalur perbekalan farmasi yang harus menyebarkan obat yang diperlukan masyarakat secara meluas dan merata.

2.10 Usability

Menurut [Jak03] "*Usability is a quality attribute that assesses how easy user interfaces are to use. The word "usability" also refers to methods for improving ease-of-use during the design process*".

Usability adalah atribut kualitas yang menjelaskan atau mengukur seberapa mudah penggunaan suatu antar muka *(interface*). Kata "*Usability*" juga merujuk pada suatu metode untuk meningkatkan kemudahan pemakaian selama proses desain.

Menurut [Lun01] '' *Usability appears to consist of usefulness and ease of use, and usefulness and ease of use are correlated. Each factor in turn drives user satisfaction and frequency of use* ".

Dalam artikelnya [Lun01] mengemukakan bahwa mereka telah mengembangkan alat untuk mengukur usability software, yakni kuesioner USE. USE terdiri dari *Usefulness* (Kegunaan), *Satisfaction* (Kepuasan), *and Ease of use* (Kemudahan Penggunaan) dan indikator *Ease of Use* dikembangkan menjadi dua yaitu *Ease of Use* dan *Ease of Learning*